Condor Compatible Tools for Data Intensive Computing

Douglas Thain University of Notre Dame

Condor Week 2011

The Cooperative Computing Lab

- We collaborate with people who have large scale computing problems in science, engineering, and other fields.
- We operate computer systems on the scale of 1200 cores. (Small)
- We conduct computer science research in the context of real people and problems.
- We publish open source software that captures what we have learned.
 http://www.nd.edu/~ccl

What is Condor Compatible?

Work right out of the box with Condor.
 Respect the execution environment.
 Interoperate with public Condor interfaces.

http://condor.cse.nd.edu



86160 (47%) CPU-Hours Unused 76483 (42%) CPU-Hours Used by Condor 18839 (10%) CPU-Hours Used by Owner 181482 (100%) CPU-Hours Total

Top Condor Users for the Last Week

	CPU Hours	Percent Total	Max Jobs Running	Max Jobs Queued
vvijayan@nd.edu	24339	31.18%	450	780
athrash1@nd.edu	16458	21.08%	501	618
pbrenne1@nd.edu	15439	19.78%	104	125
pbui@nd.edu	6423	8.23%	154	162
jthomp11@nd.edu	5494	7.04%	200	200
rcarmich@nd.edu	3652	4.68%	828	2000
lyu2@nd.edu	3398	4.35%	30	30
ccl@nd.edu	2131	2.73%	89	406

<image>

	0.0	1.52	0.02	0.01	1.95	2.05	2.03	1.01	1.03	0.29	2.09	0.1	1.02
	<u>c-104-13</u>	calvin	ccl00	ccl01	cc103	cc103	cc104	ccl05	cc106	ccl07	ccl12	coloube	celdb e
	- 7) [?	1 8 0.0	1000 17 0.16	876 <mark>142</mark> 0.13	1005 <mark>12</mark> 0.45	25 0.41	1015 02 0.15	1011 06 0.11	1008 09 0.16	1189 <mark>8</mark> 0.16	42 32 0.24	12 239 0.01	4 89 5 0 31 0
	eels, bmit00	colsubmit0*	l celsubmit02	celsubmit03	3 cclsun00	cclsun01	cclsun02	cclsun03	cclsun04	cclsun05	cclsun06	cclsun07	cclsun08 c
4	4 // 1370	4 1371 0.69	5 1369 9 24	0 1374	101 357 0.55	4 404	81 397 9 24	9 <mark>6</mark> 363	31 376 1 25	76 381 9 20	<mark>89 368</mark> 1 201	87 390 1 2 2 1	74 383 1
	eckup11	eclsup12	celsun13	celsun14	eelsun15	colweb00	colweb01	cclweb02	colweb03	celus01	cellos12	celuis03	celuis07 ci
8	388	78 384	78 384	<mark>76 382</mark>	90 367	3 <u>697</u>	96	0 <u>96</u>	0 96	3 <u>109</u>	9/ 193	/ 194	0 419
	L I I I I I I I I I I I I I I I I I I I		J.26	J.20	J 3	1.88	J.17	1.49	J.16	alava 442	207	24 markin un	0.0 1
	197	4 182	166 0	18 167			4 86	4 <u>5</u>	0 9		4 16	0 19	3 15 4
	01	1.22	0.16	0.09			0. <mark>49</mark>		0.13		0.0	0.99	1.02 0
	confucius	cse-c0-00	cse-c0-01	cse-c0-02	cse-cU-aa	cse-c0-64	cse-c0-05	cse-dcws-01	40 80	cse-ws-00	cse-ms-01	48 56	46 59 8
	1.01	0.19	1.19	5.18	0.18	0.17	0.23	0.08	0.16	0.45	0.27	0.1	0.4 6
	cse-ws-08	cse-ws-09	cse-os-10	cse-ws-11	cse-ws-19	cse-ws-20	cslewis	cvrl-c0-00	cvrl-c0-01	cvrl-c0-02	evri-c0-03	cvrl-c0-04	cvrl-c0-05 c
1	80 24 0,34	0.13	1.19	4.37	0.02	0.02	10 82	1.0	0.0	5.06	76 07	0.0	4.98 07
E	dbrt120-02	dbrt131-02	ddcopt037	denali	dickens	diocles	dintito	dint01	dint02	dinkas	dint04	dirt05	dint06 d
Ľ.	0 9 0 0	9 11.13	7 109 1 0	0.05	4 86 8.03	48 56		14 52 2 21	10 56 2 2	1 , 53	4 62 2 18	14 52 8 34	10 <u>66</u> 0 5 07 9
	disc03	disc04	disc05	disc06	disc07	disc08	discOS	disc10	disc11 🦼	disc12	disc14	disc15	disc16 d
в	0 1833 4 07	0 1833	0 1833	0 1833	0 1858	0 1833	0 1833	0 1833	1837	0 1/33	0 1833	0 1833	0 1833 0
	dise20	dise21	dise22	dise23	7.37 disc24	dise 26	dise26	dm1	durna	droutk	a.u i fab an abysis	faulkner	foster fr
в	0 1833	0 1833	0 1833	0 1833	0 1833	0 1853	0 1833	282	3 / 6	4 6	6 87	<mark>7 83</mark>	0 9 4
	4.88	4.28	0.09	0.0	0.05	4.15		101	2.06 / /	0/2	0.0	0.05	0.01 0
7P	gareng	garibaldin 0 9	garibaldi4 0 109	garibaldio 0 109	gatotkaca	gh1U1 65	gh10a		0 // 65	0112 0 65	gh114 0 65	gh110 0 65	gh117 g 0 270 0
	1.02		1.0	1.0	1.03	4.15	4.15	24 1	4,/ <u>ø</u> /	1.1 <mark>4</mark>	2.23	3.15	
	gh122	gh201	gh202	gh203	gh203	gh204	gh206	gh207	91.208	gh210	gh213	gh214	gh215 g
	0.0	2.16	4.16	2.47	2.44	1.1 <mark>4</mark>	1.1 <mark>8</mark>	A 16		1.1 <mark>5</mark>	27	1.0	2.05 1
-	gh302	gh303	gh306	gh308	gh309	gh312	gh314	gh315	gh316	gh317	gh319	gh320	gh322 g
	2.84	3.16	0.16	0 270	0.0 270	1.0	2.57		1 1 1 1 1 1 1 1 1 1	2/0	1.01 270	0.0 270	1.01
_	gh325	gh326	gh327	gh332	gh337	gh339	gh340	gran db o eku	pylieg	head	hegel	hepinc193	hepinc317 h
0	0 270	0 270	0 270	0 270	0 270	0 270	0 270	3697 / 68 V	6	8 10 1 17	2 2	363	3/2 4
	joemuffaw	kant	krisna	laotse	larson	lcg1164	liszt	logizi	roahler	maimonide	s maritain	meyrury	meurer m
5	15 0	4 5	4 5	4 5	4 86	238		A 143/ A	<u>al 114</u>	4 5	3 6	85	9/ 111 8
	nakula	netscale01	netscale02	netscale03	netscale04	netscale()8	Deiscale 10	netscale11	netspale12	netscale 13	netscale 14	ne scale 151	aetscale16 p
5	7 2	241	3 241	3 240	8 241	8 878	878	//878	7 878	878	8 878	878	878
	1.97	U.96	1.97 romoto2	1.72	U.U	0.0	0.0	0.0 ~~5.00	0.0	40.02	0.0	20.04	0.0
4	5 4	5 23	0 8	0 194	4 274	4 5	40 07	X9 7	836 X	651 - 6	810 8	1.67	581 6 5

And the "challenging" users... I submitted 10 jobs yesterday, and that worked, so I submitted 10M this morning! And then I write the output into 10,000 files of 1KB each. Per job. Did I mention each one reads the same input file of 1TB? Sorry, am I reading that file twice? What do you mean, sequential access? Condor is nice, but I also want to use my cluster, and SGE, and Amazon and... 5

Idea:

Get the end user into telling us more about their data needs.

In exchange, give workflow portability and resource management.

Makeflow



part1 part2 part3: input.data split.py ./split.py input.data

out1: part1 mysim.exe ./mysim.exe part1 >out1

out2: part2 mysim.exe ./mysim.exe part2 >out2

out3: part3 mysim.exe ./mysim.exe part3 >out3

result: out1 out2 out3 join.py ./join.py out1 out2 out3 > result

Douglas Thain and Christopher Moretti, <u>Abstractions for Cloud Computing with Condor</u>, Syed Ahson and Mohammad Ilyas, *Cloud Computing and Software Services: Theory and Techniques*, pages 7 153-171, CRC Press, July, 2010.

Makeflow = Make + Workflow



Abstract System Interface





Michael Albrecht, Patrick Donnelly, Peter Bui, and Douglas Thain, Makeflow: A Portable Abstraction for Cluster, Cloud, and Grid Computing. 9



Andrew Thrasher, Rory Carmichael, Peter Bui, Li Yu, Douglas Thain, and Scott Emrich, <u>Taming Complex Bioinformatics Workflows with Weaver, Makeflow, and Starch,</u> *Workshop on Workflows in Support of Large Scale Science*, pages 1-6, November, 2010

Weaver

Weaver Code
jpgs = [str(i)+'. jpg ' for i in range (1000)]
conv = SimpleFunction('convert',out_suffix ='png ')
pngs = Map(conv,jpgs)

```
# Makeflow Code
0.png: 0.jpg /usr/bin/convert
      /usr/bin/convert 0.jpg 0.png
1.png: 1.jpg /usr/bin/convert
      /usr/bin/convert 1.jpg 1.png
...
999.png: 999.jpg /usr/bin
```

```
/usr/bin/convert 999.jpg 999.png
```

Peter Bui, Li Yu and Douglas Thain, <u>Weaver: Integrating Distributed Computing Abstractions into Scientific</u> <u>Workflows using Python</u>, *CLADE*, June, 2010.

Makeflow and Work Queue



Makeflow and Work Queue



SAND - Scalable Assembler



Christopher Moretti, Michael Olson, Scott Emrich, and Douglas Thain, <u>Highly Scalable Genome Assembly on Campus Grids</u>, *Many-Task Computing on Grids and Supercomputers (MTAGS)*, November, 2009

Replica Exchange on WQ

Replica Exchange

Work Queue API





Connecting Condor Jobs to Remote Data Storage

Parrot and Chirp

Parrot – A User Level Virtual File System

- Connects apps to remote data services:
- HTTP, FTP, Hadoop, iRODS, XrootD, Chirp
- No special privileges to install or use.
- Chirp A Personal File Server
 - Export existing file services beyond the cluster.
 Local disk, NFS, AFS, HDFS
 - Add rich access control features.
 - No special privileges to install or use.





Patrick Donnelly, Peter Bui, Douglas Thain, <u>Attaching Cloud Storage to a Campus Grid Using Parrot, Chirp, and Hadoop</u>, *IEEE Cloud Computing Technology and Science*, pages 488-495, November, 2010.

Putting it All Together



Computer Science Challenges

- With multicore everywhere, we want to run multiple apps per machine, but the local OS is still very poor at managing resources.
- How many workers does a workload need? Can we even tell when we have too many or too few?
- How to automatically partition a data intensive DAG across multiple multicore machines?
- \$\$\$ is now part of the computing interface. Does it make sense to get it inside the workflow and/or API?

What is Condor Compatible? Work right out of the box with Condor. makeflow –T condor - condor submit workers Respect the execution environment. - Accept eviction and failure as normal. Put data in the right place so it can be cleaned up automatically by Condor. Interoperate with public Condor interfaces. - Servers run happily under the condor master. – Compatible with Chirp I/O via the Starter. 23

A Team Effort

- Faculty:
 - Patrick Flynn
 - Scott Emrich
 - Jesus Izaguirre
 - Nitesh Chawla
 - Kenneth Judd

- Grad Students Undergrads
 - Hoang Bui
 - Li Yu
 - Peter Bui
 - Michael Albrecht Zach Musgrave
 - Patrick Donnely
 - Peter Sempolinski
 - Dinesh Rajan

- - Rachel Witty
 - Thomas Potthast
 - Brenden Kokosza

 - Anthony Canino

NSF Grants CCF-0621434, CNS-0643229, and CNS 08-554087.

For More Information

The Cooperative Computing Lab – <u>http://www.nd.edu/~ccl</u>

Condor-Compatible Software: – Makeflow, Work Queue, Parrot, Chirp, SAND

Prof. Douglas Thain – <u>dthain@nd.edu</u>